

REMARKS

Claims 1-24 are pending in the Application, and all claims stand rejected by the Office Action mailed October 2, 2008. Claims 1, 3, 16, and 22 are amended by this response. Claims 1, 16, and 22 are independent claims. Claims 2-15, 17-21, and 23-24 depend from independent claims 1, 16, and 22, respectively.

Applicants respectfully request reconsideration of pending claims 1-24, in light of the following remarks.

Objection to Declaration

Based on a telephone conversation Monday, December 8, 2008, between Examiner Zhen and Applicants' representative Kevin Borg, Applicants understand that the objection to the declaration has been withdrawn, and a new declaration is no longer required. Applicants appreciate the Examiner's time and cooperation in resolving this issue.

Rejection of Claims

Rejections under 35 U.S.C. §101

Claims 1-21 were rejected under 35 U.S.C. §101. The Office Action asserts that the claimed invention is directed to non-statutory subject matter. As detailed further below, Applicants respectfully traverse the rejection.

Claims 1 and 16 (as well as their dependent claims) stand rejected on the grounds that the "service broker" is a "software component," and that a "software component is a computer program that is merely a set of instructions capable of being executed by a computer." (See Office Action at p. 2.) Applicants respectfully submit that the "service broker" is not claimed as a computer listing *per se*, and is therefore a statutory invention.

MPEP § 2106.01(I) provides,

Similarly, computer programs claimed as computer listing *per se*, i.e., the descriptions or expressions of the programs, are not physical "things." They are neither

computer components nor statutory processes, as they are not "acts" being performed. Such claimed computer programs do not define any structural and functional interrelationships between the computer program and other claimed elements of a computer which permit the computer program's functionality to be realized. In contrast, a claimed computer-readable medium encoded with a computer program is a computer element which defines structural and functional interrelationships between the computer program and the rest of the computer which permit the computer program's functionality to be realized, and is thus statutory. See *Lowry*, 32 F.3d at 1583-84, 32 USPQ2d at 1035. Accordingly, it is important to distinguish claims that define descriptive material *per se* from claims that define statutory inventions.

Applicants respectfully submit that the "service broker" is not a computer listing *per se*. The MPEP recites "computer listings *per se*" as "the descriptions or expressions of the programs." In contrast, the MPEP discusses statutory inventions as, for example, "a computer element which defines structural and functional interrelationships between the computer program and the rest of the computer which permit the computer program's functionality to be realized, and is thus statutory." Applicants respectfully submit that the "service broker" as claimed is not limited to any one particular service broker, but may include, for example, a server such as service broker server 127, as pointed out previously. Applicants respectfully submit that such a server, as understood by one of skill in the art, would include a computer element defining structural and functional interrelationships between a computer program and the rest of a computer that would permit a computer program's functionality to be realized, and would thus be statutory. (See *also* claim 11, which expressly recites "a server...")

With regard to claim 3, the Office Action asserts that "claim 3 specifically identifies the service broker as a software component. Therefore, the system as recited in claims 1 and 16 includes a service broker that is a software component." (Office Action at p. 2.) The doctrine of claim differentiation teaches that if claim 3, *arguendo*, recites a computer listing *per se*, then, by definition, independent claim 1 is not necessarily limited only to that computer listing *per se* purportedly recited in a claim dependent therefrom. Moreover, claim 3 has been amended (see below) to clarify that it recites the system of claim 1

wherein the service broker comprises a software component in the electronic device. (Further, claim 16 is independent from both claims 1 and 3.) In any event, a computer program may be a part of the service broker (which may, for example, also include a server, and/or also include a computer readable medium), but the service broker itself is not a computer listing *per se*, as explained by the MPEP:

Computer programs are often recited as part of a claim.
USPTO personnel should determine whether the computer program is being claimed as part of an otherwise statutory manufacture or machine. In such a case, the claim remains statutory irrespective of the fact that a computer program is included in the claim. The same result occurs when a computer program is used in a computerized process where the computer executes the instructions set forth in the computer program. **Only when the claimed invention taken as a whole is directed to a mere program listing, i.e., to only its description or expression, is it descriptive material *per se* and hence nonstatutory.**

(MPEP § 2106.01(I); emphasis added). Here, a computer program may be part of the "service broker," but the "service broker" itself is an otherwise statutory manufacture or machine. That the "service broker" may contain a computer program does not transform the entire service broker into only a computer listing -- the service broker, taken as whole, is not directed to a "mere program listing." As such, Applicants respectfully submit that the service broker, as claimed, defines a statutory invention, and that claims 1-16 (and their dependent claims) are allowable over §101.

While Applicants disagree with the Office Action's assertion that the "software component" of claim 3 is a computer listing *per se*, in the interests of expediting prosecution, Applicants have amended claim 3 to recite the system of claim 1 wherein the service broker comprises a software component in the electronic device. Again, any computer listing that may be included in the subject matter recited by claim 3 (which depends from claim 1, which defines statutory subject matter as discussed above) is being claimed as part of an otherwise statutory manufacture or machine. Applicants respectfully submit that claim 3 is allowable over §101.

Rejections under 35 U.S.C. §103

Claims 1-24 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Publication No. 20030084138 (hereinafter "Tavis") in view of U.S. Patent No. 6,202,207 (hereinafter "Donohue"). Applicants respectfully submit that the Office Action has failed to establish a *prima facie* case of obviousness, and further respectfully traverse the rejections.

Tavis and Donahue Do Not Render Obvious Claims 1-15

Claim 1 has been amended for the purpose of clarification. Amended claim 1 recites "[a] system that facilitates interactions between one of a plurality of software components in an electronic device and an associated one of a plurality of servers, via a network, the system comprising: a service broker capable of receiving at least one request for service associated with one of the plurality of software components, wherein the request for service does not identify a location from which to obtain the service; the service broker capable of determining the one of the plurality of servers associated with the one of the plurality of software components, based upon a prior registration associating the one of the plurality of servers with the one of the plurality of software components making the at least one request for service; and the service broker capable of forwarding the at least one request for service to the determined one of the plurality of servers." Support for the amended language (namely, "wherein the request for service does not identify a location from which to obtain the service") may be found in the specification at, for example, paragraph [0040]: "In one embodiment of the present invention, a client-side software component, such as the applications component 113, may desire an update to its software from a service provider, although it may not know to which service provider to communicate a request." Claims 16 and 22 have been amended in a generally similar manner.

The Office Action asserts that Tavis teaches a service broker "capable of receiving at least one request for service associated with one of the plurality of software components [Every component update requests [sic] begins when some object calls the component with a component URL; paragraphs 0086 and 0047]." (See Office Action at

p. 4.) Applicants respectfully submit that the teachings of Tavis, as well as the Office Action's characterizations of those teachings, are contradictory to the presently claimed subject matter.

For example, the Office Action characterizes Tavis as teaching that "[e]very component update requests [sic] begins when some object calls the component with a component URL." Thus, the location of the "component" being requested (the "component URL") is included in the request. This is quite different, and patently distinct, from the presently claimed subject matter, including "wherein the request for service does not identify a location from which to obtain the service."

The cited portions of Tavis confirm this distinction. Paragraph 0086 reads as follows:

Every component update requests begins [sic] when some object calls the component with a component URL. This first component URL 560 is identical in format to the component URLs (for example, URLs 536 and 544) that point from one OSD file to another. However, component URL 560 is special because it specifies the root of the component hierarchy from which the chain of trust will be established. It is also special in that it is not a secure component URL. The steps for establishing this initial trust are illustrated in FIG. 6. The process begins in step 600 and proceeds to step 602 in which a component manager, having determined that the component specified by the initial component URL needs to be installed, uses the download manager to download the OSD file 500 specified in that URL 560.

(emphasis added.) Again, Applicants note that the update request "begins when some object calls the component with a component URL." In contrast, in the presently claimed subject matter, the request for service does not identify a location from which to obtain the service. The other portion of Tavis cited with respect to this aspect of the presently claimed subject matter is paragraph 0047, which reads as follows:

More particularly, there are at least four ways to trigger the update process. These include a user generated

update request in which a user (or process) generates a request by performing one of several actions. For example, a user could start the collaborative system for the first time, the user could add a new or updated tool to a shared space or the user could invite a new user to join a shared space. In the case of a user generated update request, a local component update delta is generated and then dispatched to other shared space members. When a new user is invited to join a shared space, the new member receives a copy of the shared space which incorporates a list of required components. These are both examples of a synchronous notification.

Thus, this cited portion of Tavis adds nothing to the previously discussed portion, and is silent with respect to a request for service that does not identify a location from which to obtain the service.

In fact, taken as a whole, Tavis not only does not teach this aspect of the presently claimed subject matter, but also actually teaches against it, as the inclusion of the URL in the request of Tavis would teach against using a request that did not identify a location. Further, Tavis explicitly states that the URL is always included:

A component update request that a component manager receives asks for a component resource by providing a URL that identifies the resource to the component manager. Component resource URLs always point to the Internet, via an HTTP or an FTP scheme. The format of a component resource URL has two parts separated by a "?" character. The following is a simple example:....

(Tavis at [0056]; emphasis added). As Tavis expressly states that the component update request received by the component manager includes a URL identifying a resource, and that the component resource URL always points to the internet, Tavis therefore expressly teaches against a request that does not identify a location from which to obtain the service.

Moreover, Tavis, taken as a whole, teaches against other aspects of the presently claimed subject matter, including "the service broker capable of determining the one of the plurality of servers associated with the one of the plurality of software components..." (See also previous submissions regarding the failure of Tavis to teach "the service broker capable of determining...") It would not make sense, for example, to incorporate a service broker capable of determining the one of the plurality of servers into the system of Tavis, where every request already identifies a component to be called by URL. With the location already specified in Tavis, the Office Action does not explain what is left to "determine." Once the request includes the location, there would be no reason, for example, to include a service broker capable of determining the one of the plurality of servers. As such, Applicants respectfully submit that one skilled in the art would not be motivated to arrive at the presently claimed subject matter, including "wherein the request for service does not identify a location from which to obtain the service" or "the service broker capable of determining...", and, in fact, Tavis teaches against the presently claimed subject matter. As a result of the foregoing, Applicants respectfully submit that Tavis and Donahue, either alone or in combination, do not teach, suggest, or otherwise render obvious these aspects of the presently claimed subject matter, and that claim 1 and its dependent claims are allowable for at least those reasons.

Further, Applicants respectfully submit that the cited art does not teach, suggest, or otherwise render obvious "the service broker capable of forwarding the at least one request for service to the determined one of the plurality of servers," as recited by claim 1. In asserting that Tavis teaches this aspect of the presently claimed subject matter, the Office Action states that Tavis teaches, "the service broker capable of forwarding the at least one request for service to the determined one of the plurality of servers [download manager 130 retrieves components designated by the system component manager 116 from a server over a network, such as the Internet; p. 3, paragraph 0034 and p. 6, paragraph 0076]." (See Office Action at p. 5.)

As an initial matter, the asserted teaching (namely, "download manager 130 retrieves components designated by the system component manager 116...") is quite

different from, and does not teach, "forwarding the at least one request for service to the determined one of the plurality of servers." Applicants respectfully submit that the "forwarding the at least one request..." is patentably distinct from "retriev[ing] components." In the presently claimed subject matter, a request for service is forwarded to a server. In contrast, in the teaching of Tavis as asserted by the Office Action, a component is retrieved. Further, in the teaching of Tavis as interpreted and asserted by the Office Action, the system component manager 116 (the purported service broker) designates the components retrieved, whereas in the presently claimed subject matter, the service broker forwards a request for service to the determined one of the plurality of servers. Applicants respectfully submit that the retrieval of Tavis as asserted by the Office Action is quite different from, and does not teach, suggest, or otherwise render obvious, for example, the "forwarding the at least one request for service to the determined one of the plurality of servers."

An examination of the cited portions of Tavis confirms that Tavis does not teach this aspect of the presently claimed subject matter. Tavis at [0034] reads as follows:

The system component managers 116 works in conjunction with a download manager 130 and an install manager 132. The download manager 130 retrieves components designated by the system component manager 116 from a server over a network, such as the Internet and the install manager 132 installs retrieved components in the local system. The system component manager 116 communicates with the download manager 130 via the download request queue 118, and communicates with the install manager 132 via the install request queue 120.

As also stated by Tavis in the following paragraph, "...As discussed below, each component is identified by a Uniform Resource Locator (URL) that is passed to the download manager 130 by the system component manager 116 to begin the download process..." (Tavis at [0035]). Such retrieval of an identified component is silent with respect to, and does not teach "...forwarding the at least one request for service..." as presently claimed. The second cited portion of Tavis, [0076] reads as follows:

After calling the transport interface, the download manager thread reads from the stream pending, if necessary and writes the stream data to a file in the download manager staging area. The name of the file is determined by the components OSD name and the staging area is partitioned by the URL path. The download manager also notifies the system component manager of its progress, via an element queue, if requested. Finally, when the download manager thread receives an end-of-stream return code, it closes the staging file and notifies the system component manager that the file transfer, but not the file verification, is complete. In a preferred embodiment, components can be downloaded from "component farms", which are commodity servers that host components available for download by the component manager.

Again, the "downloads" of this portion of Tavis are quite different from, and do not teach "...forwarding the at least one request for service..." as presently claimed. Applicants respectfully submit that retrieving a download is patentably distinct from forwarding a request. (See also, for example, Specification at [0040]: "...The service broker server 127 may then forward the received software update request to one of the appropriate service providers, such as the service provider A 129, which, in turn, may process the received request for a software update, retrieve an update package and associated information, and communicate the update package and associated information back to the mobile handset 107..."")

As a result of the foregoing, Applicants respectfully submit that the cited art does not teach, suggest, or otherwise render obvious the subject matter claimed by claim 1, and that the Office Action does not present a *prima facie* case of obviousness. Applicants further respectfully submit that claim 1, and its dependent claims 2-15, are therefore allowable over the cited art.

Tavis and Donahue Do Not Render Obvious Claims 16-21

As indicated above, claim 16 has been amended in a manner generally similar to claim 1 to clarify certain aspects of its claimed subject matter. Applicants respectfully submit that independent claim 16 recites elements similar in many ways to those recited by independent claim 1, and that the Office Action cites many of the same portions of the cited art in the rejection of claim 16, as were cited in the rejection of claim 1. Accordingly, Applicants respectfully submit that independent claim 16 is allowable over the proposed cited combination, for at least the reasons set forth above with respect to the rejection of independent claim 1. Further, because claims 17-21 depend from allowable independent claim 16, Applicant respectfully submits that claims 17-21 are also allowable, for at least the same reasons. Accordingly, Applicant respectfully requests that the rejection of claims 16-21 under 35 U.S.C. §103(a) be reconsidered and withdrawn.

Tavis and Donahue Do Not Render Obvious Claims 22-24

As indicated above, claim 22 has been amended in a manner generally similar to claim 1 to clarify certain aspects of its claimed subject matter. Applicants respectfully submit that independent claim 22 recites elements similar in many ways to those recited by independent claim 1, and that the Office Action cites many of the same portions of the cited art in the rejection of claim 22, as were cited in the rejection of claim 1. Accordingly, Applicants respectfully submit that independent claim 22 is allowable over the proposed cited combination, for at least the reasons set forth above with respect to the rejection of independent claim 1. Further, because claims 23-24 depend from allowable independent claim 16, Applicant respectfully submits that claims 23-24 are also allowable, for at least the same reasons. Accordingly, Applicant respectfully requests that the rejection of claims 22-24 under 35 U.S.C. §103(a) be reconsidered and withdrawn.

Conclusion

In general, the Office Action makes various statements regarding the claims and the cited references that are now moot in light of the above. Thus, the Applicant will not address such statements at the present time. However, the Applicant expressly reserves the right to challenge such statements in the future should the need arise (e.g., if such statements should become relevant by appearing in a rejection of any current or future claim).

The Applicant believes that all of pending claims 1-24 are in condition for allowance. Should the Examiner disagree or have any questions regarding this submission, the Applicant invites the Examiner to telephone the undersigned at (312) 775-8000.

A Notice of Allowability is courteously solicited.

Respectfully submitted,

Dated: December 19, 2008

/Kevin E. Borg/
Kevin E. Borg
Reg. No. 51,486

Hewlett-Packard Company
Intellectual Property Administration
Legal Department, M/S 35
P.O. Box 272400
Fort Collins, CO 80527-2400